

## 錬成問題

- 以下に示すのは、今日（プログラム実行時）の日付を表示するプログラムである。

```
import java. (1) .GregorianCalendar; •———— 1
import (2) java. (1) .GregorianCalendar.*; •———— 2

class Today {
    (3) static (4) main(String[] args) {
        GregorianCalendar today = new GregorianCalendar(); •———— 3

        System.out.printf("今日は%04d年%02d月%02d日です。 \n",
            today.get((5)), today.get((6)) + 1, today.get((7)));
    }
}
```

型名の表記には2種類がある。フルネーム表記である `java. (1) .GregorianCalendar` は、(8) と呼ばれ、単なる `GregorianCalendar` は、(9) と呼ばれる。(3)の宣言において (8) ではなく (9) を利用できるのは、1の宣言があるからである。このような宣言を、“(10)宣言”と呼ぶ。

また、クラスの“(11)メンバ”を(9)で表せるようにするのが、2の宣言である。この宣言を、“(12)宣言”と呼ぶ。

ここで宣言されているクラス `Today` が所属するのは“(13)パッケージ”である。その(8)は(14)であり、(9)は(15)である。

- 平方根を求める `Math.sqrt` メソッドや円周率を表す変数 `Math.PI` を、単なる `sqrt` や `PI` でアクセスするためには、以下の宣言が必要である。

```
import (16);
```

- 日付を表す `GregorianCalendar` クラスと乱数を生成する `Random` クラスを、(8)でなく(9)である `GregorianCalendar` や `Random` でアクセスするためには、以下の宣言が必要である。

```
import (17);
```

- `Math` クラスが所属するパッケージは(18)であり、`System` クラスが所属するパッケージは(19)である。

- `public` なクラス `abc` と `public` でないクラス `xyz` の両方を格納するソースファイルの名前は(20)とする。クラス `abc` は(21)、`xyz` は(22)る。

- ▶ (21)・(22)の選択肢：(a)同一パッケージでのみ利用でき  
(b)他のパッケージからのみ利用でき  
(c)任意のパッケージから利用でき

- 三つのクラス  $X$ ,  $Y$ ,  $Z$  が以下のように宣言されている。

```
package a; 4
public class X {
    public void m1() { /* ... */ }
    protected void m2() { /* ... */ }
    void m3() { /* ... */ }
    private void m4() { /* ... */ }

    void f1(X a) {
        // ...
    }
}
```

```
package a;
public class Y {
    void f2(X a) {
        // ...
    }
}
```

```
package b;
import a.X;
class Z {
    void f3(X a) {
        // ...
    }
}
```

メソッド  $f1$ ,  $f2$ ,  $f3$  から呼び出すことのできるメソッドには○を、呼び出せないメソッドには×を記入せよ。

- ◆ メソッド  $f1$  からの呼出し:

$m1$  ...  (23)     $m2$  ...  (24)     $m3$  ...  (25)     $m4$  ...  (26)

- ◆ メソッド  $f2$  からの呼出し:

$m1$  ...  (27)     $m2$  ...  (28)     $m3$  ...  (29)     $m4$  ...  (30)

- ◆ メソッド  $f3$  からの呼出し:

$m1$  ...  (31)     $m2$  ...  (32)     $m3$  ...  (33)     $m4$  ...  (34)

4の宣言は、“ (35) 宣言”と呼ばれる。この宣言は、一つのソースプログラム中に (36) 置ける。また、1や2の宣言 (37)。

- ▶  (36) の選択肢: (a)何個でも (b)0個または1個 (c)1個
- ▶  (37) の選択肢: (a)とは共存できない (b)とは無関係にどこにでも置ける (c)より後ろに置く必要がある (d)より前に置く必要がある

■ クラスのメンバ・コンストラクタのアクセス性には4種類がある。アクセス性はキーワードを付けて指定する。キーワード **public**, **protected**, **private** が指定するのは、それぞれ“ (38) アクセス”、“ (39) アクセス”、“ (40) アクセス”である。

なお、クラス自体が **public** でない場合、**public** 付きで宣言されたメンバ・コンストラクタのアクセス性は、“ (41) アクセス”となる。

また、キーワードを指定しないメンバ・コンストラクタのアクセス性は、“ (42) アクセス”あるいは“ (43) アクセス”と呼ばれる。